**What is OS:**

* A collection of programs which deal directly with the hardware system and sub-systems, provides user interfaces and logs the activities taking place in the system.
* It is the software that enables all the programs we use.
* The OS organizes and controls the hardware.
* OS acts as an interface between the application programs and the machine hardware.
* Examples: Windows, Unix, Linux, Mac OS etc...

**What does an OS do?**

An operating system performs basic tasks such as

* Controlling and allocating memory.
* Prioritizing system requests.
* Controlling input and output devices.
* Facilitating networking .
* Managing file systems.
* Memory management
* Resource monitoring

* The evolution of OS's went through seven major phases:
* Six of them significally changed the ways in which users accessed computers through the open shop, batch processing, multiprogramming, timesharing, personal computing and distributed systems
* In the seventh phase the foundations of concurrent programming(a way of using a whole lot of expert systems working together) were developed and demonstrated in model OS's.

**Batch processing:**

* In batch processing same type of jobs batch (BATCH-a set of jobs with similar needs) together and execute at a time.
* The OS was simple, its major task was to transfer control from one job to the next.
* The job was submitted to the computer operator in form of punch cards. At some later time the output appeared.
* The OS was always resident in memory
* Common input devices were card readers and tape drives
* Common output devices were line printers, tape drives and card punches.
* Users did not interact directly with the computer systems, but he prepared a job (comprising of the program, the data & some control information

**Multiprogramming:**

* Multiprogramming is a technique to execute number of programs simultaneously by a single processor.
* In multiprogramming, number of processes reside in the main memory at a time.
* The OS picks and begins to executes on of the jobs in the main memory.
* If any I/O wait happened in a process, then CPU switches from that job to another job.
* Hence CPU is not idle at any time.

**Advantages:**

* Efficient memory utilization
* CPU is never idle so performance increases

**Time Sharing system**

* Time Sharing or multitasking is a logical extension of multiprogramming.
* Multiple jobs are executed by switching the CPU between them
* In this, the CPU time is shared by different processes, so it is called as "Time sharing systems".
* Time slice is defines by the OS, for sharing CPU time between processes
* Examples: Multics, Unix etc.

**OS functions:**

1. Program Creation
2. Program Execution
3. Input/output operations
4. Error detection
5. Resource allocation
6. Accounting (keeps account/track of different types on files. Calculates the number of bytes that are being used and what is left.)
7. Protection

**Types of OS:**

Single user Systems:

* Provides a platform for only one user at a time.
* They are popularly associated with desk top OS's which run on standalone systems where no user accounts are required e.g. DOS

Multi User Systems:

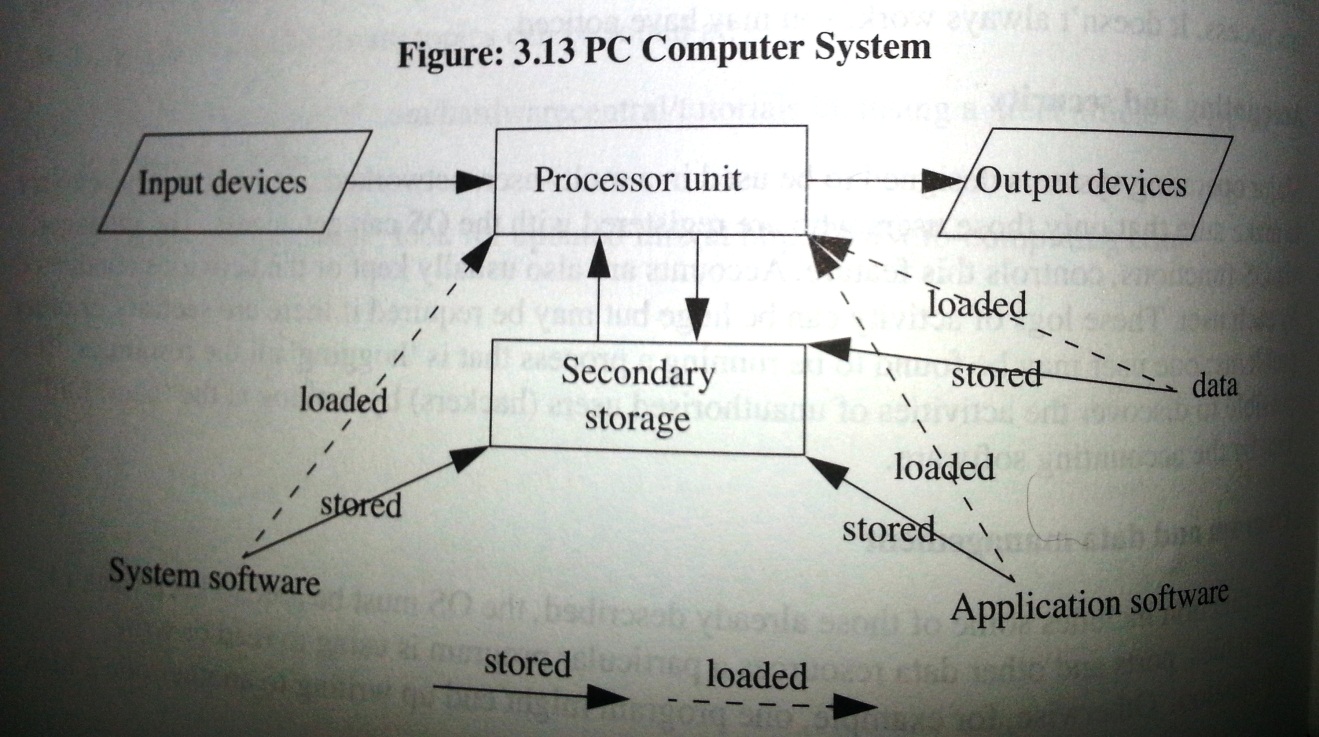
* Provides regulated access for a number of users by maintaining a database of known users.
* Refers to computer systems that support 2 or more simultaneous users.
* Another term for multi-user is time sharing.
* Ex: Unix
* Ex: all mainframes are multi user systems

**Characteristics of Computer Systems:**

* Computers are made up of input, processing and output communications hardware devices and the systems and application software required to operate and connect(interface) the components so they can function and communicate
* Computers systems can consist of single computers such as a PC or can be composed of a number of computers linked together into a large network
* Computers can vary in size of memory & secondary storage & the speed of the processor

**Single computers consist of:**

* Input devices
* Processing devices
* Output and communication devices
* Backing store
* System storage
* System software
* Application software
* Stored data



**Communicating with Peripherals:**

The operating system (OS) manages the peripherals (ex. printer, graphics card, keyboard, etc.) using drivers

Driver = piece of software, that interacts directly with the hardware.

The OS provides a standard interface between hardware devices and applications.

Enables generic commands such as the 'print'-statement within the high-level programming languages to work on any PC, even if the hardware differs greatly.

Hardware is only able to deal with binary machine code or if a conversion between digital and analog formats has taken place (digital to analog conversion).

Examples of applications requiring analogue to digital conversion: speech recognition, temperature sensing, light detection

Plug and play is a feature of the operating system that allows devices to be used without installing extra drivers.

**Coordinating Concurrent Processing:**

There are constantly processes running whenever the computer is turned on. The OS manages/handles the loading and unloading of processes to & from primary memory.